

**Práctica Universitaria Conducente a Trabajo de Grado. Grupo De Investigación GIA**

Diego Fernando Calvera Hernandez

Director De La Práctica: Julio Cesar López Betancur

Universidad Tecnológica De Pereira

Facultad De Ingenierías

Programa Ingeniería De Sistemas Y Computación

Pereira - Colombia

2021

## Contenido

1.	Introducción.....	5
2.	Objetivos.....	5
2.1	Objetivo General.....	5
2.2	Objetivos Específicos .....	5
3.	Planteamiento Del Problema .....	6
4.	Justificación .....	6
5.	Marco Teórico Y Conceptual .....	7
5.1	Acerca De Metodología Ágil Scrum .....	7
5.2	Acerca De Framework.....	8
5.3	Acerca De Las Herramientas De Desarrollo .....	9
5.3.1	Acerca De Spring.....	9
5.3.2	Acerca De PlantUml. ....	10
5.3.3	Acerca De Netbeans.....	10
5.3.4	Acerca De Markdown. ....	11
6.	Marco Metodológico .....	12
7.	Proceso De Desarrollo .....	12
8.	Bibliografía.....	35

## Índice De Figuras

Ilustración 1, Diagrama de clase de entidades del módulo PlanAccion .....	14
Ilustración 2, Diagrama de clase Aya .....	14
Ilustración 3, Diagrama de clase del módulo Authentication .....	15
Ilustración 4, Diagrama de casos de uso del módulo de Authentication .....	15
Ilustración 5, Diagrama de actividades del módulo Authentication .....	16
Ilustración 6, Diagrama de clases del módulo Bonificacionesbsc .....	17
Ilustración 7, Diagrama de clases de servicios del módulo Bonificacionesbsc .....	17
Ilustración 8, Diagrama de casos de uso de las bonificaciones .....	18
Ilustración 9, Diagrama de casos de uso de costos .....	19
Ilustración 10, Diagrama de casos de uso de empleados .....	19
Ilustración 11, Diagrama de casos de uso de novedad .....	20
Ilustración 12, Diagrama de casos de uso de rango .....	20
Ilustración 13, Diagrama de clases del submódulo empresa .....	21
Ilustración 14, Diagrama de clases del submódulo de fuente de datos .....	21
Ilustración 15, Diagrama de casos de uso de empresa .....	22
Ilustración 16, Diagrama de casos de uso de fuente de datos .....	22
Ilustración 17, Diagrama de clases 1 de módulo Correo-server .....	23
Ilustración 18, Diagrama de clases 2 de módulo Correo-server .....	23
Ilustración 19, Diagrama de clases de iniciativa y portafolio .....	24
Ilustración 20, Diagrama de clases de avance y actividad .....	25
Ilustración 21, Diagrama de clases de entidades del módulo Iniciativas .....	25
Ilustración 22, Diagrama de clases del módulo Plan-estratégico-back .....	26
Ilustración 23, Diagrama de clases de los Tipos Rae .....	26

Ilustración 24, Diagrama de clases de Asistentes .....	27
Ilustración 25, Diagrama de clases de Compromisos .....	27
Ilustración 26, Diagrama de clases de Gestión Rae y Archivos .....	28
Ilustración 27, Diagrama de clases de Gestión Categoría Rae .....	28
Ilustración 28, Diagrama de clases del módulo Registrar-valores-back .....	29
Ilustración 29, Diagrama de casos de uso de Valores Categóricos .....	29
Ilustración 30, Diagrama de casos de uso de Feing.....	30
Ilustración 31, Diagrama de clases 1 de módulo Usuarios .....	30
Ilustración 32, Diagrama de clases 2 de módulo Usuarios .....	31
Ilustración 33, Diagrama de casos de uso de módulo Usuarios .....	32
Ilustración 34, Diagrama de clases Vis categoría del módulo de Visualización .....	33
Ilustración 35, Diagrama de clases Vis indicador del módulo de Visualización .....	33
Ilustración 36, Diagrama de casos de uso de Vis categoría del módulo de Visualización .....	34
Ilustración 37, Diagrama de casos de uso de Vis indicador del módulo de Visualización .....	34

## **1. Introducción**

La práctica universitaria empresarial estuvo orientada a reforzar la aplicación de los procesos de ingeniería de Software (documentación, levantamiento de requisitos, realización de diagramas, etc.) y las metodologías en desarrollo, soporte y mantenimiento de software en el grupo de Investigación GIA UTP, de acuerdo con las líneas de investigación y extensión que se trabajan en el grupo de investigación.

## **2. Objetivos**

### **2.1 Objetivo General**

Participar en el desarrollo, soporte y mantenimiento de aplicativos que requiere el grupo de investigación para el cumplimiento de sus objetivos académicos en investigación y extensión.

### **2.2 Objetivos Específicos**

Capacitación en herramientas de desarrollo.

Aplicación de la ingeniería de software en la práctica universitaria.

Aplicación de metodologías ágiles en el proceso de desarrollo de software.

### **3. Planteamiento Del Problema**

El grupo de investigación GIA UTP adscrito al programa de Ingeniería de sistemas y computación viene trabajando en líneas de investigación y extensión, donde constantemente requiere la aplicación de la ingeniería de software, metodologías para el desarrollo de software, planeación de proyectos y administración de proyectos de software para cumplir con los objetos contractuales con las diferentes entidades con las cuales tiene convenios.

Por lo cual las actividades realizadas durante el proceso de las practicas fueron:

Cursos de capacitación en herramientas de desarrollo, uso de metodologías de desarrollo ágil, aplicación de los procesos de la ingeniería de software (levantamiento de requerimientos, diseño de software, arquitectura, desarrollo, soporte y mantenimiento).

### **4. Justificación**

Durante el desarrollo de software se ha evidenciado la gran importancia que tiene el uso de los frameworks, facilitando el proceso de desarrollo de aplicaciones, centrando la atención verdaderamente en el problema y no en las funcionalidades que son de uso común, como el registro de usuarios, la conexión con base de datos, entre otros.

Del mismo modo se ve reflejado el proceso de documentación en los proyectos de software, facilitando la comprensión del código, posibilitando la adaptación del software para su mantenimiento y futuras actualizaciones.

A su vez, la importancia del uso de la metodología Scrum, el cual es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente y obtener el mejor resultado posible de proyectos, además, resalta e impulsa el trabajo en equipo, el aprendizaje constante y una estructura que es flexible a los cambios que van sucediendo en la fase de desarrollo.

Es así cómo se realiza en la práctica universitaria empresarial, el proceso de aprendizaje de la herramienta Spring Framework para el entendimiento de los proyectos de software realizados por el grupo de investigación GIA, y de esta manera, trabajando colaborativamente, realizar la documentación pertinente del proyecto BSC.

## **5. Marco Teórico Y Conceptual**

En el presente capítulo se definen los términos fundamentales para la comprensión de las expresiones usadas en el presente documento.

### **5.1 Acerca De Metodología Ágil Scrum**

Es un marco de trabajo utilizado dentro de equipos de trabajo que manejan proyectos complejos, es una metodología de trabajo ágil, con el fin de realizar entregas en periodos cortos de tiempo, basado en tres pilares fundamentales: la transparencia, la cual indica que todos los implicados en el proyecto tienen conocimiento de que ocurre y como ocurre, teniendo un entendimiento común del proyecto. La adaptación, la cual se refleja al momento de implementar

cambios, el equipo se ajusta para lograr el objetivo del “sprint”. Y la inspección, en la cual los miembros del equipo frecuentemente inspeccionan el progreso para detectar posibles problemas.

En Scrum existen 3 roles muy importantes: Product Owner, responsable de hablar constantemente con el cliente, por lo tanto, debe tener mucho conocimiento sobre el negocio. Scrum Master, el cual cumple la labor de ayudar en la adopción de esta metodología en todos los equipos, es un líder que se encarga de eliminar inconvenientes que tenga el equipo dentro de un Sprint (contiene todas las tareas del proceso de desarrollo del software, con cada sprint el equipo entrega una parte funcional del producto completo), aplicando las mejores técnicas. Y el Equipo de desarrollo, los cuales son los encargados de realizar las tareas priorizadas por el Product Owner.

## **5.2 Acerca De Framework**

Los frameworks se caracterizan por ser un grupo de herramientas que, acompañados de una estructura, permiten el desarrollo de aplicaciones de forma fácil, escalable y mantenible.

Principalmente facilita el desarrollo ágil, partiendo de una base común para nuestros proyectos. Esto nos permite reducir los tiempos de desarrollo, así como beneficiarnos de un código sólido y de buena calidad, esto gracias a que se trata de desarrollos de gran confianza y con gran soporte por parte de la comunidad.

Se trata de una manera muy optimizada de trabajar y que permite unificar la metodología de trabajo dentro de un equipo de desarrollo, ofreciendo seguridad, mantenibilidad y eficiencia a los proyectos realizados.



### **5.3 Acerca De Las Herramientas De Desarrollo**

A continuación, se presenta la definición y características de las herramientas usadas en el desarrollo de las prácticas universitarias empresariales.

#### **5.3.1 Acerca De Spring.**

Spring es un framework, el cual hace que la programación de Java sea más rápida, fácil y segura para todos. El enfoque de Spring en la velocidad, la simplicidad y la productividad lo ha convertido en el marco de trabajo Java más popular del mundo.

Spring ofrece experiencias agradables a millones de usuarios finales todos los días, ya sea transmisión de TV, compras en línea u otras innumerables soluciones innovadoras. Spring también tiene contribuciones de todos los grandes nombres de la tecnología, incluidos Alibaba, Amazon, Google, Microsoft y más.

Spring tiene un historial probado en el manejo de problemas de seguridad de manera rápida y responsable. Los responsables de Spring trabajan con profesionales de seguridad para parchear y probar cualquier vulnerabilidad reportada. Las dependencias de terceros también se controlan de cerca y se emiten actualizaciones periódicas para ayudar a mantener sus datos y aplicaciones lo más seguros posible.

### **5.3.2 Acerca De PlantUml.**

PlantUML es una herramienta de código abierto que permite a los usuarios crear diagramas a partir de un lenguaje de texto sin formato. Además de varios diagramas UML, PlantUML tiene soporte para varios otros formatos relacionados con el desarrollo de software

Utiliza el software Graphviz para diseñar sus diagramas y Tikz para la compatibilidad con LaTeX. Las imágenes se pueden generar como PNG, SVG, LaTeX e incluso arte ASCII. PlantUML también se ha utilizado para permitir que las personas ciegas diseñen y lean diagramas UML.

### **5.3.3 Acerca De Netbeans.**

NetBeans comenzó como un proyecto de estudiantes (originalmente llamado Xelfi) en la República Checa, en 1996. El objetivo era escribir un IDE Java (entorno de desarrollo integrado) similar a Delphi en Java.

Xelfi fue el primer IDE de Java escrito en Java, con sus primeros prelanzamientos en 1997, fue un proyecto divertido para trabajar, especialmente porque el espacio IDE de Java era un territorio inexplorado en ese momento.

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. La mayoría de los desarrolladores reconocen el IDE de NetBeans como el IDE Java original. NetBeans IDE proporciona soporte para varios idiomas (PHP, JavaFX, C / C ++, JavaScript, etc.) y marcos de trabajo.

NetBeans es un proyecto de código abierto dedicado a proporcionar productos de desarrollo de software sólidos (el NetBeans IDE y la plataforma NetBeans) que abordan las necesidades de los desarrolladores, usuarios y empresas que dependen de NetBeans como base para sus productos; en particular, para permitirles desarrollar estos productos de forma rápida, eficiente y sencilla aprovechando las fortalezas de la plataforma Java y otros estándares relevantes de la industria.

#### **5.3.4 Acerca De Markdown.**

Markdown es una herramienta de conversión de texto a HTML para escritores web, permite escribir usando un formato de texto plano fácil de leer y escribir, y luego convertirlo a XHTML (o HTML) estructuralmente válido.

Esta herramienta fue creada en 2004 por John Gruber, y se distribuye de manera gratuita bajo una licencia BSD.

Aunque en realidad Markdown también se considera un lenguaje que tiene la finalidad de permitir crear contenido de una manera sencilla de escribir, y que en todo momento mantenga un diseño legible, así que para simplificar se puede considerar Markdown como un método de escritura.

## **6. Marco Metodológico**

Se utilizó la metodología de desarrollo ágil Scrum para el proceso de documentación del proyecto BSC, teniendo tres reuniones semanales con el docente guía de la práctica, para dialogar avances y asignaciones de labores al equipo de trabajo, haciendo de éste un proceso eficaz, mostrando rápidos resultados en los entregables propuestos.

A su vez se aplicaron los procesos de ingeniería de software para el realizar la respectiva documentación por medio del uso de la herramienta PlantUML, como lo fueron los diagramas de clase, diagramas de caso de uso, diagrama de secuencia, de cada uno de los módulos del proyecto BSC, y de igual manera se usó el lenguaje demarcado Markdown para el proceso de la creación de los diferentes README incluidos en el proyecto.

## **7. Proceso De Desarrollo**

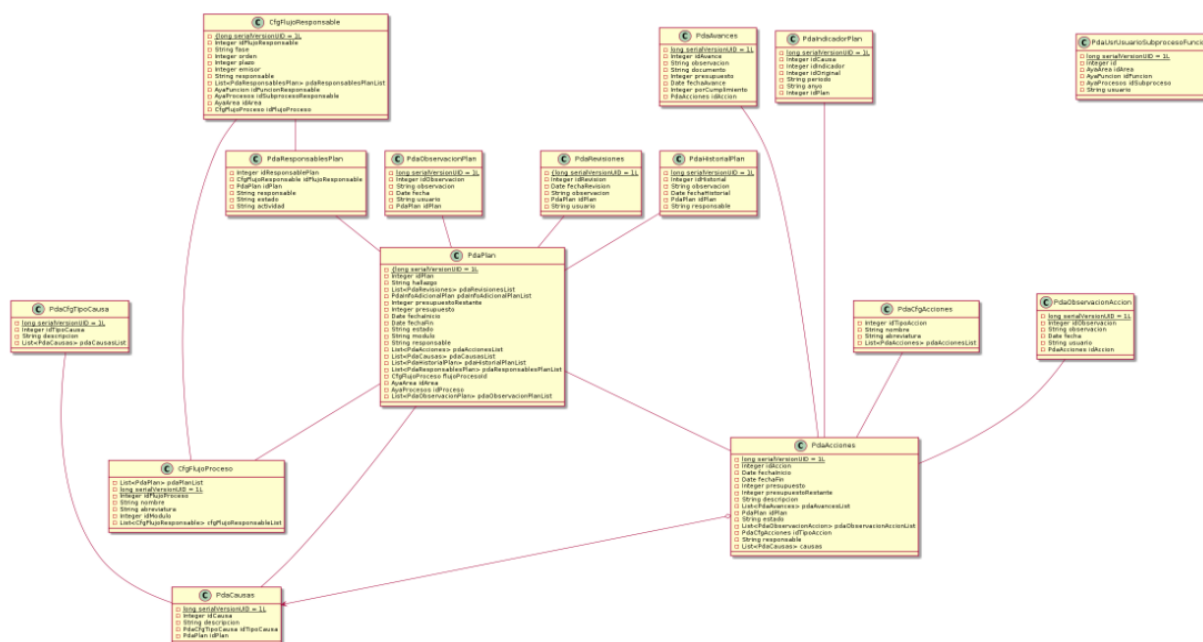
El proceso de desarrollo se basó en la metodología ágil SCRUM, dividiendo las labores entre cada uno de los practicantes (4 en total), los cuales destinaron en parejas para dos diferentes proyectos realizados por el grupo de investigación GIA.

Una vez asignados los participantes en los proyectos, en el cual se me asignó el proyecto BSC (Balanced Scorecard), se procede a realizar la inducción en el uso de la herramienta GitLab para adjuntar las versiones de los avances realizados al proyecto.

La selección de las diferentes herramientas de desarrollo usadas durante el proceso de la práctica fue realizado por el ingeniero en jefe del proyecto, quien por experiencia y conocimiento tomó la decisión de implementar el desarrollo de la documentación con la herramienta PlantUML, el cual facilita la creación de los diagramas de clase, diagramas de casos de uso y diagramas de secuencia realizados en cada uno de los diferentes módulos del proyecto; a su vez se hace uso del lenguaje demarcado Markdown para la documentación de cada uno de los módulos, facilitando la integración de los diagramas previamente realizados con PlantUML y los diferentes estilos implementados en la escritura de los archivos Readme.

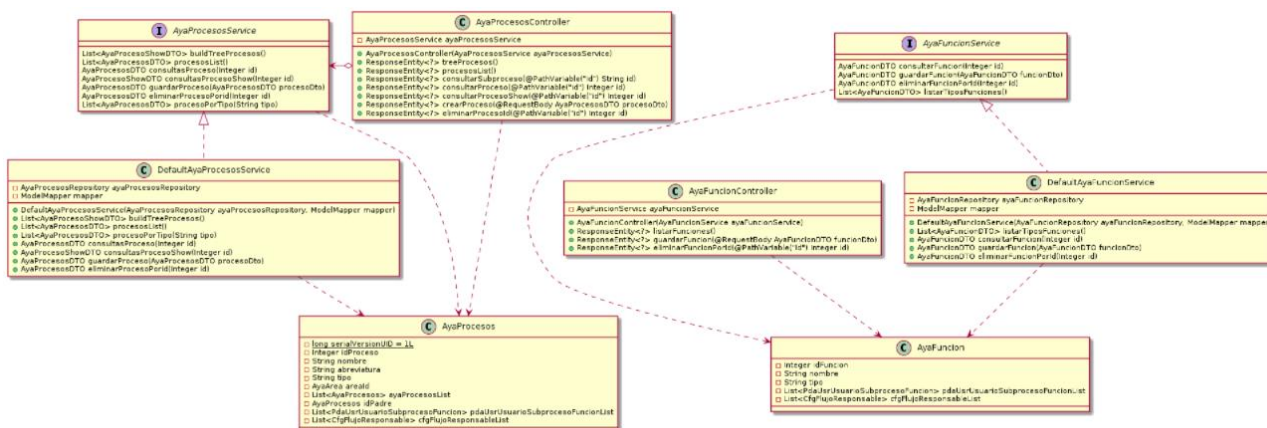
Los módulos a los cuales se les realizó el proceso de desarrollo de ingeniería de software del proyecto BSC, durante las prácticas empresariales fueron los siguientes:

**PlanAccion**, encargado de la gestión de los diferentes tipos de acciones de la base de datos, flujos de procesos, flujos responsables, causas y sus tipos, avances, historial de planes de acciones y sus responsables, observaciones y la información de usuarios.



*Ilustración 1, Diagrama de clase de entidades del módulo PlanAccion*

En el mismo módulo, existe un submódulo llamado Aya, con sus respectivos entities, services, controllers, el cual permite a los usuarios el manejo de las áreas, procesos y funciones que manejan estas mismas áreas.



*Ilustración 2, Diagrama de clase Aya*

**Authentication**, encargado de la autenticación, búsqueda y gestión de los usuarios del proyecto.

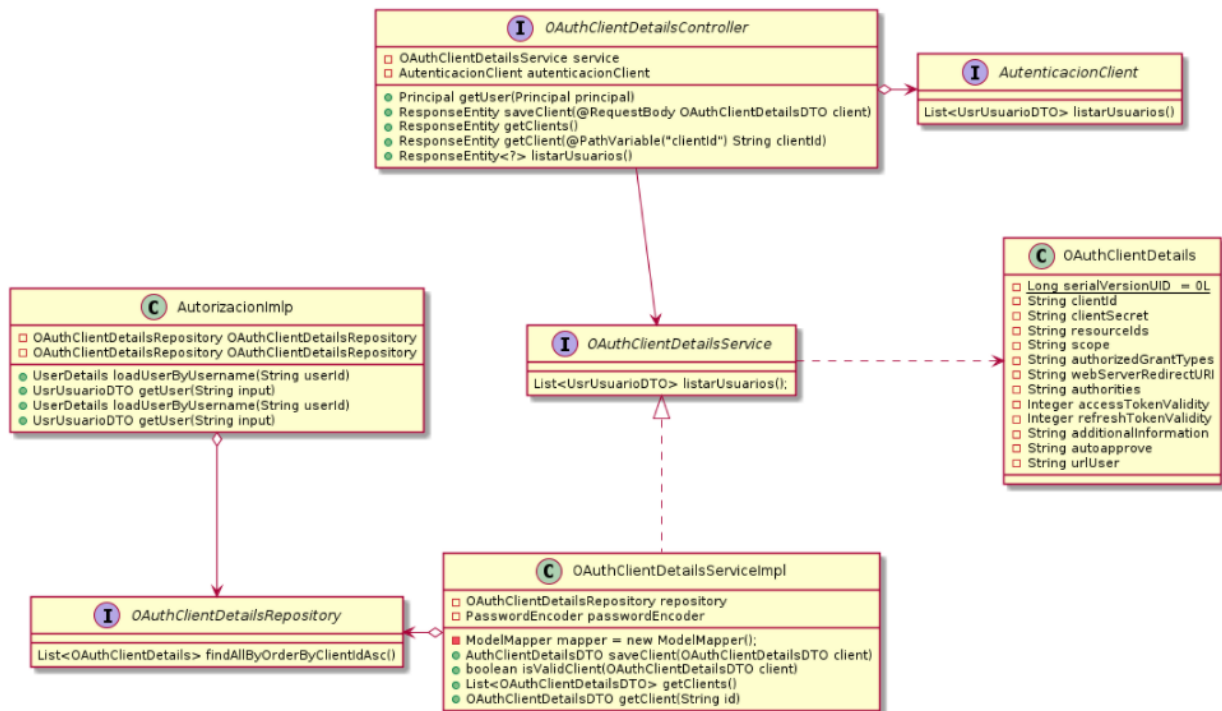


Ilustración 3, Diagrama de clase del módulo Authentication

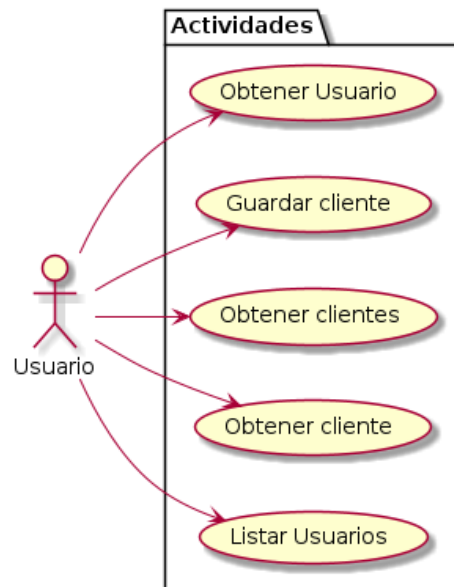
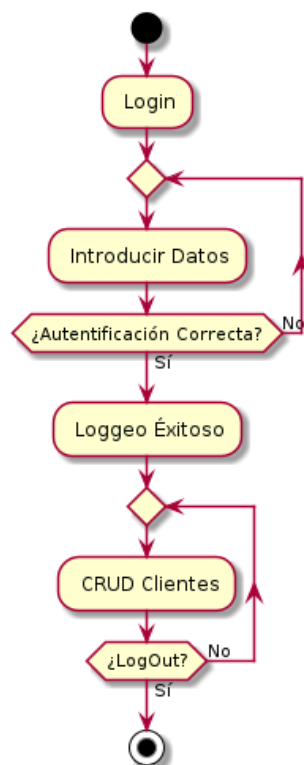


Ilustración 4, Diagrama de casos de uso del módulo de Authentication

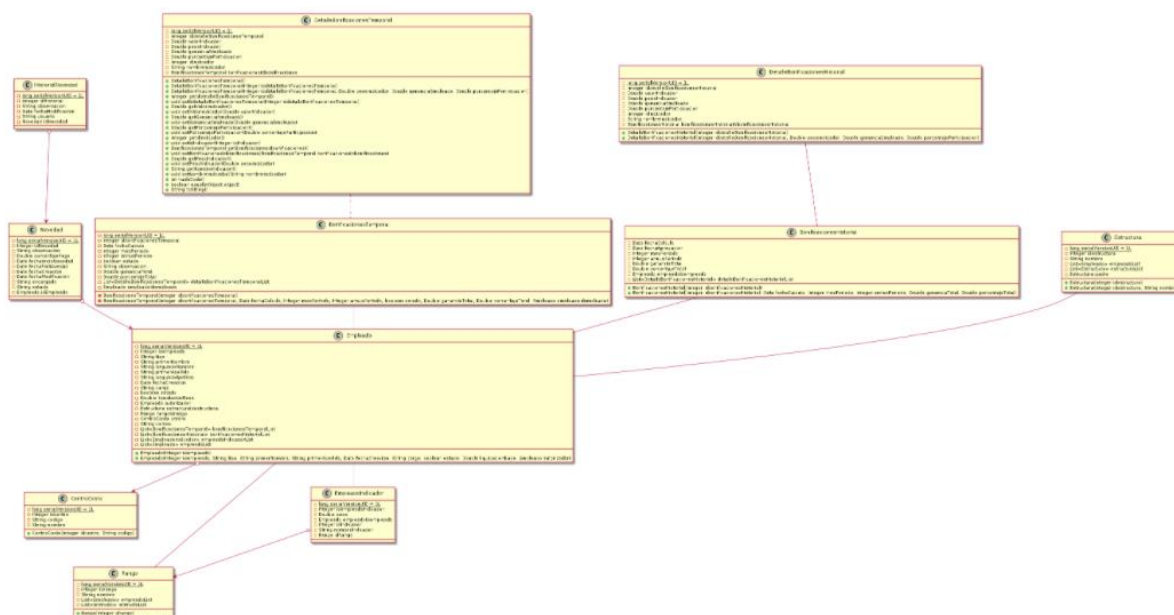


*Ilustración 5, Diagrama de actividades del módulo Authentication*

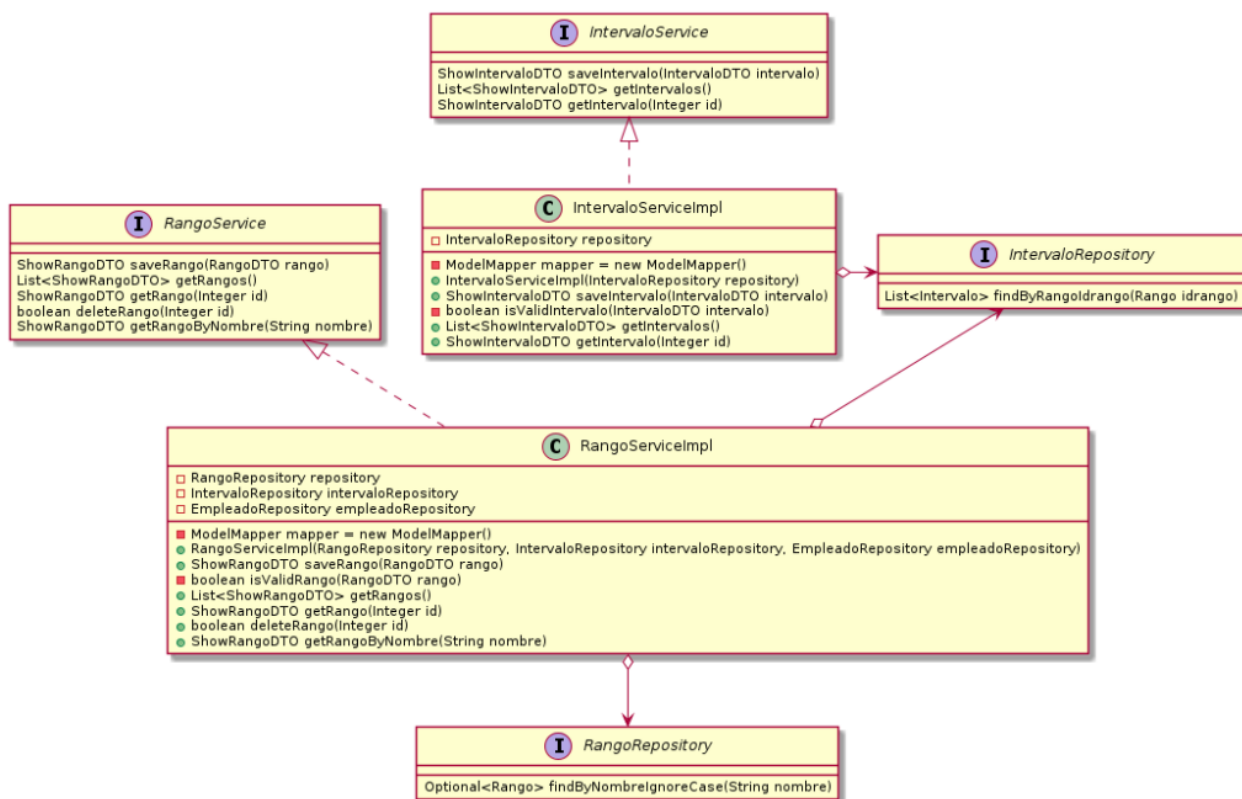
En este diagrama se busca documentar como es el proceso de login, a través de la autenticación de datos.

**Bonificacionesbsc**, encargado de la gestión de los datos del empleado, indicadores, rangos, intervalos del empleado, bonificaciones y su historial, detalles temporales y de los empleados a los que tiene a cargo.

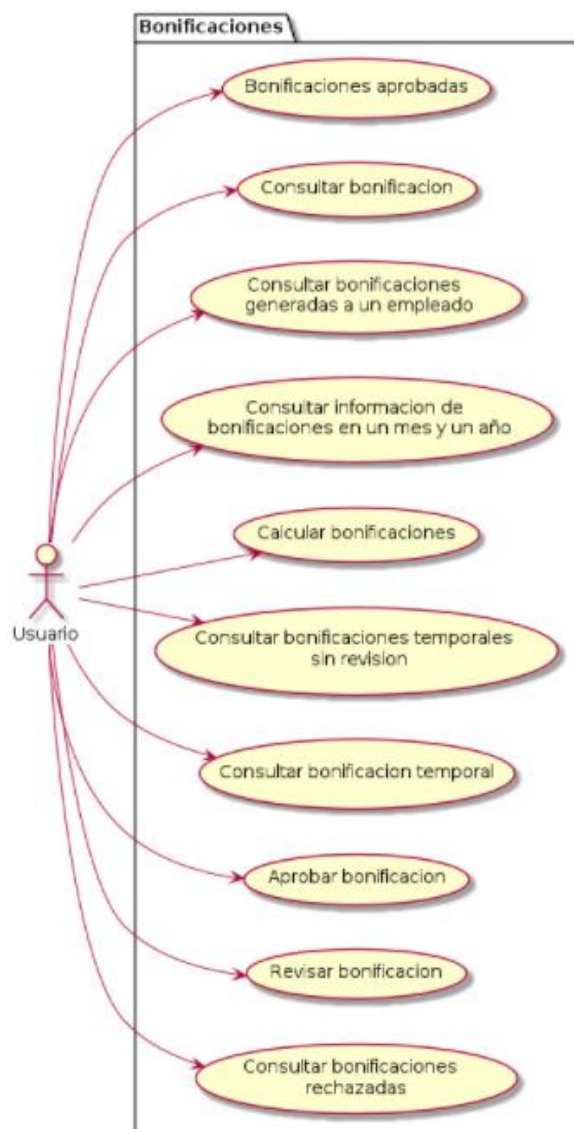




*Ilustración 6, Diagrama de clases del módulo Bonificacionesbsc*



*Ilustración 7, Diagrama de clases de servicios del módulo Bonificacionesbsc*



*Ilustración 8, Diagrama de casos de uso de las bonificaciones*

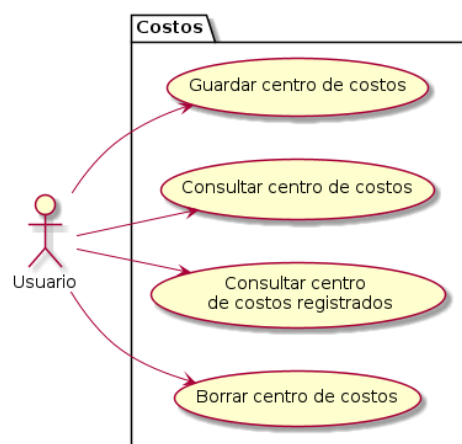
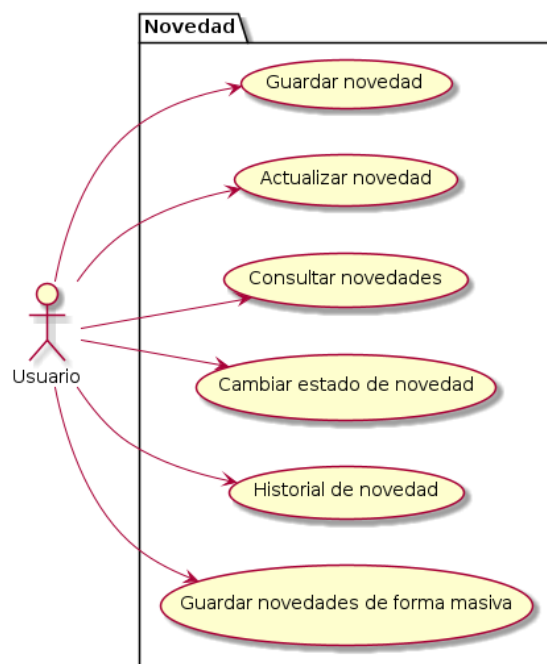


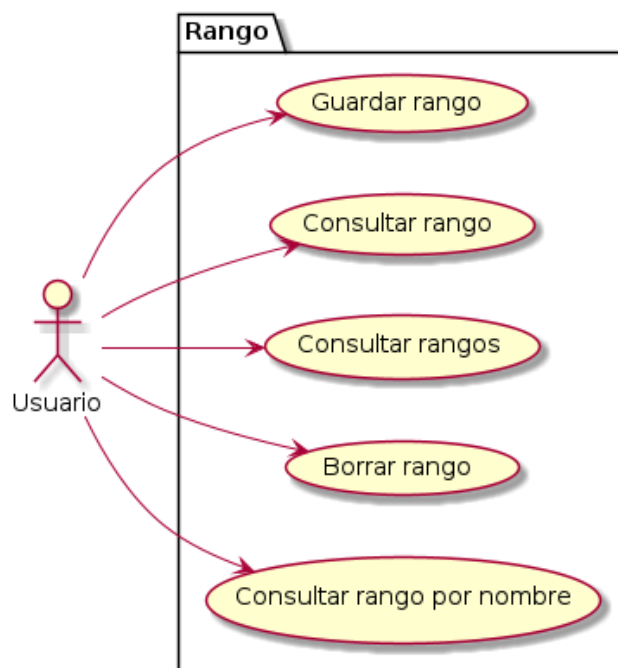
Ilustración 9, Diagrama de casos de uso de costos



Ilustración 10, Diagrama de casos de uso de empleados

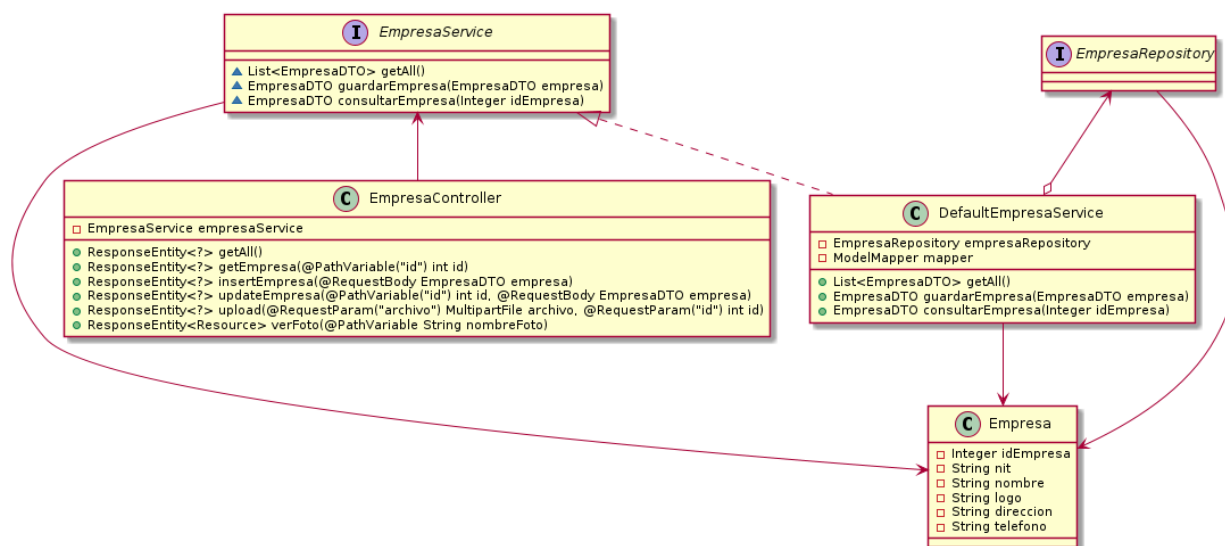


*Ilustración 11, Diagrama de casos de uso de novedad*

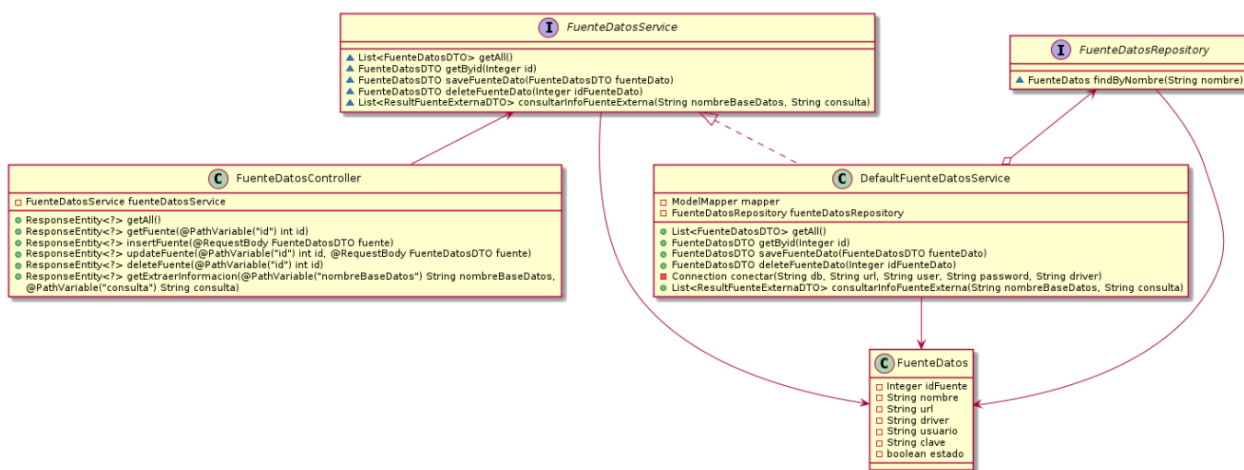


*Ilustración 12, Diagrama de casos de uso de rango*

**Config-general-back**, módulo encargado de la gestión de los datos de la empresa y la fuente de los datos del proyecto.



*Ilustración 13, Diagrama de clases del submódulo empresa*



*Ilustración 14, Diagrama de clases del submódulo de fuente de datos*

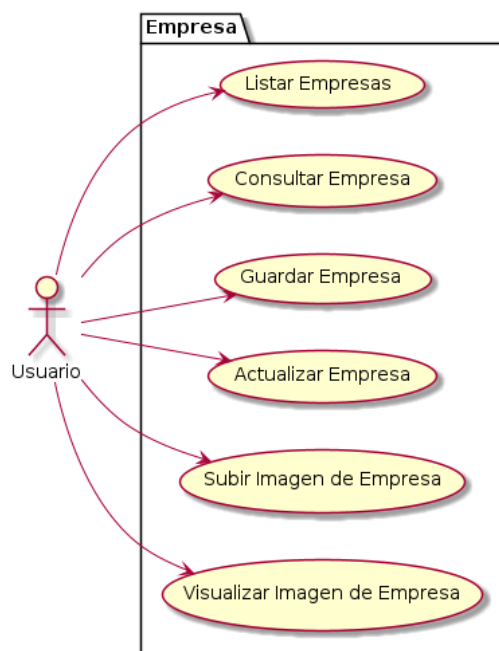


Ilustración 15, Diagrama de casos de uso de empresa

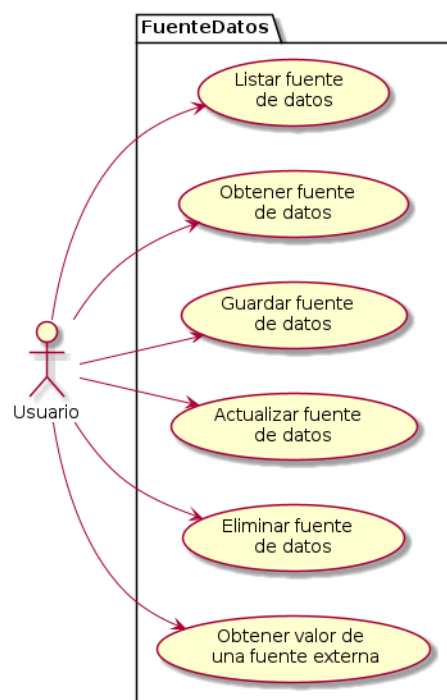


Ilustración 16, Diagrama de casos de uso de fuente de datos

**Correo-server**, encargado de la gestión de la información del correo electrónico, verificación del funcionamiento del servidor y del envío de notificaciones sobre las metas y valores.

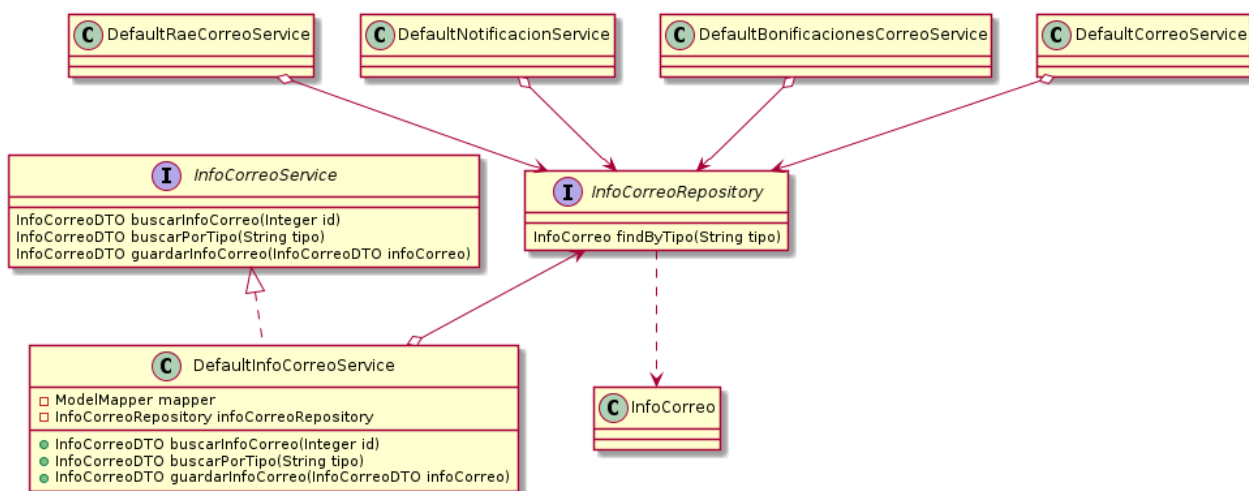


Ilustración 17, Diagrama de clases 1 de módulo Correo-server

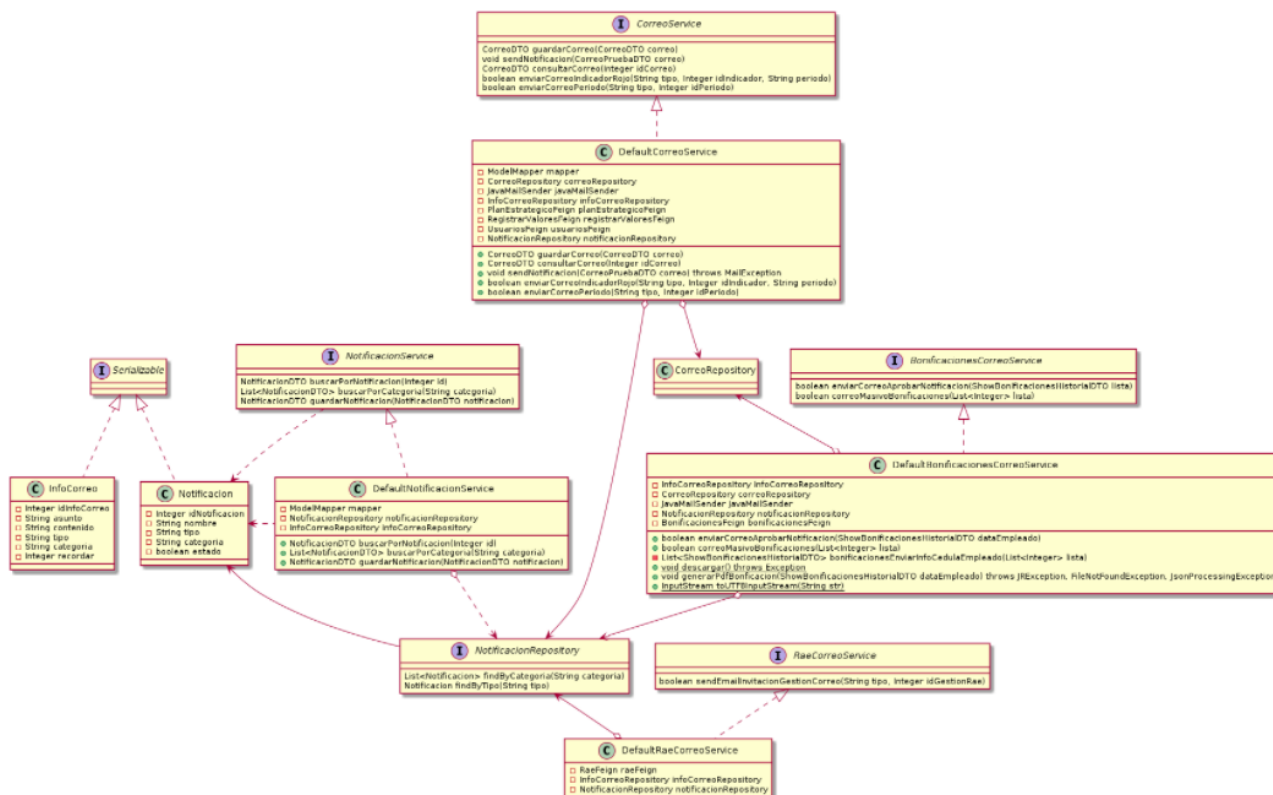
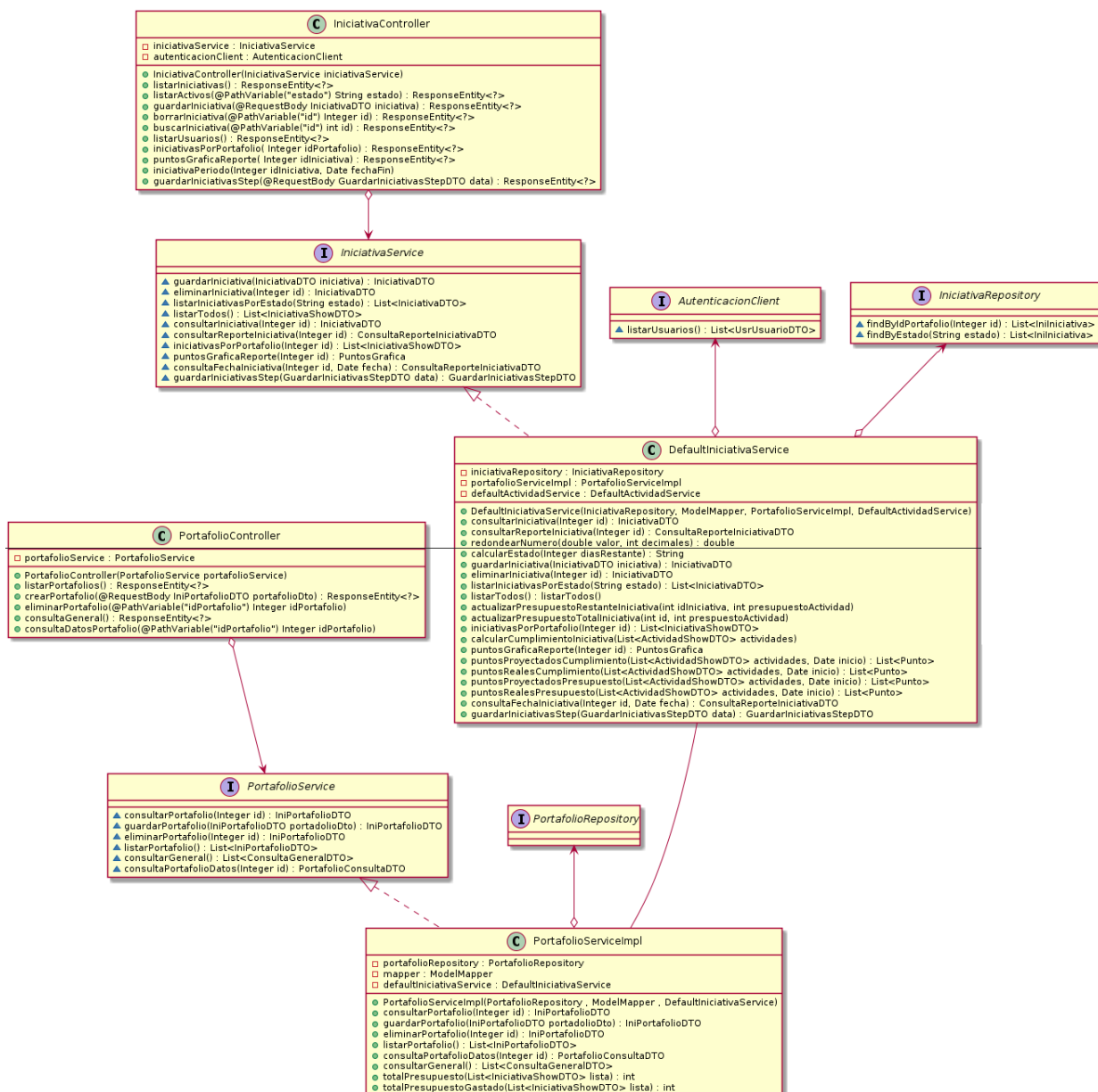


Ilustración 18, Diagrama de clases 2 de módulo Correo-server

**Iniciativas**, módulo encargado de la gestión de las iniciativas, de los avances, de las actividades y de los portafolios del proyecto.



*Ilustración 19, Diagrama de clases de iniciativa y portafolio*



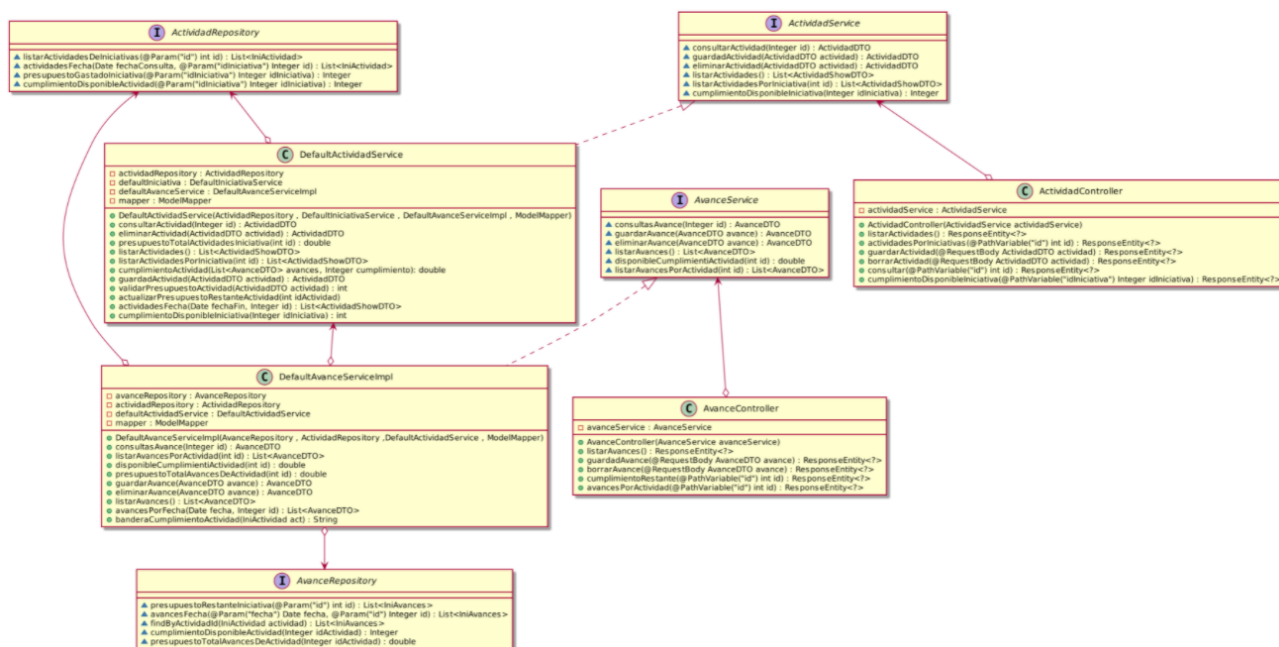


Ilustración 20, Diagrama de clases de avance y actividad

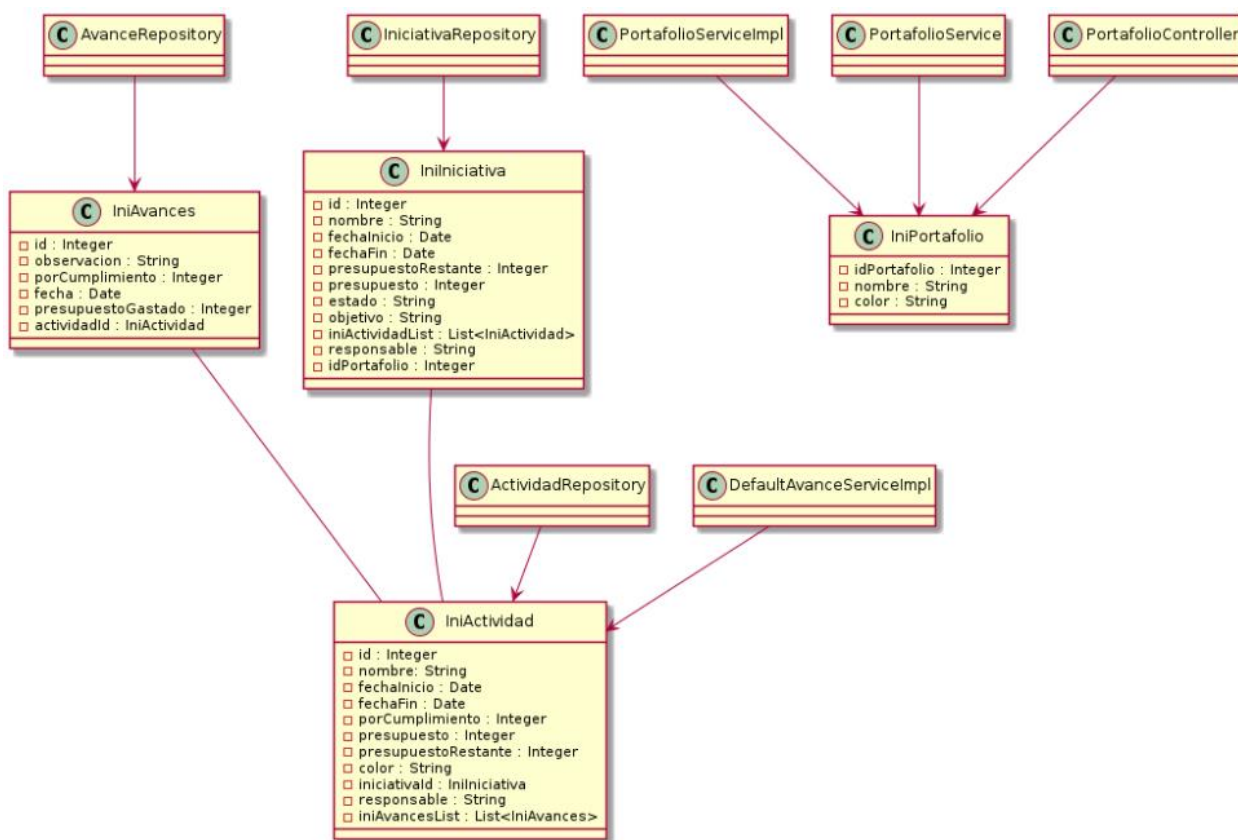


Ilustración 21, Diagrama de clases de entidades del módulo Iniciativas



**Rae-back (reunión de análisis estratégico)**, módulo encargado de la gestión de los archivos rae, asistentes pertenecientes a una rae, compromisos asociados a una rae, categorías de las rae, gestiones rae y su respectivo estado, y de los tipos de rae.



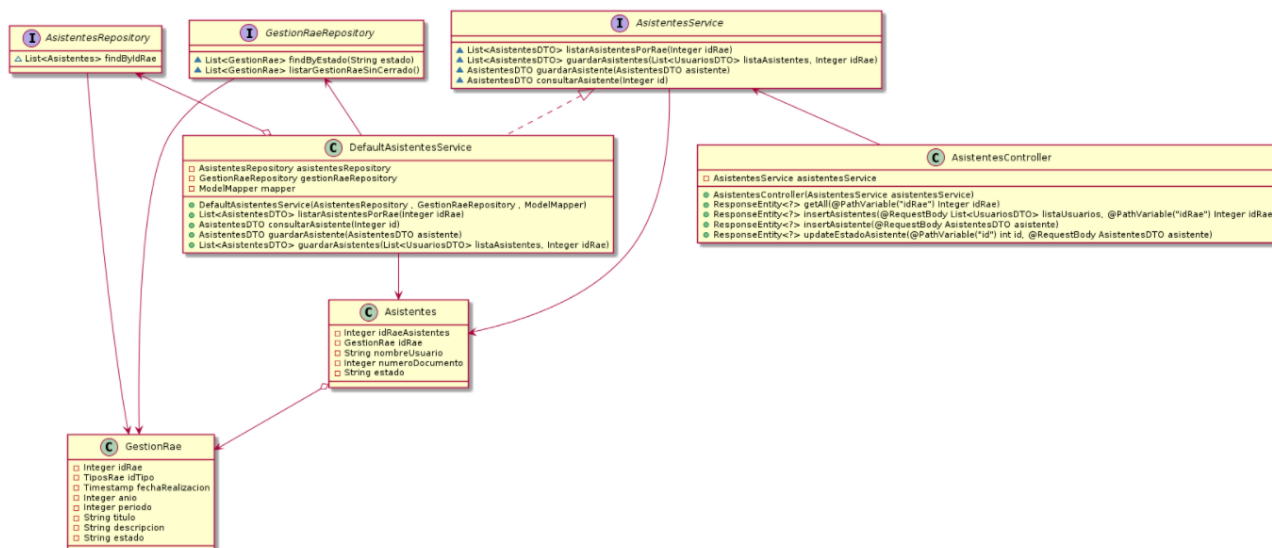


Ilustración 24, Diagrama de clases de Asistentes

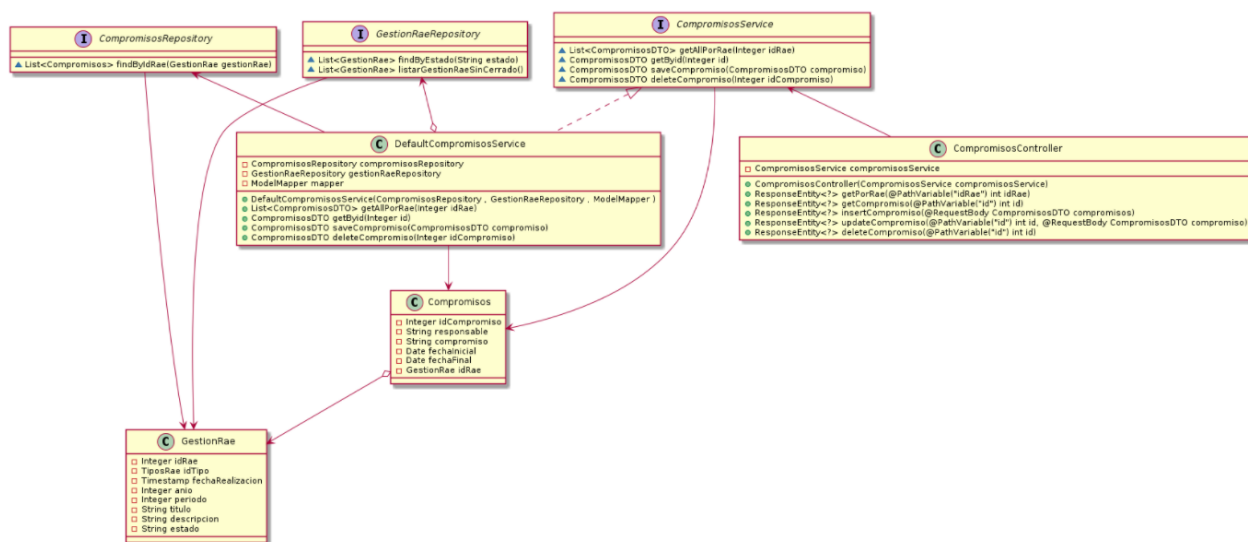
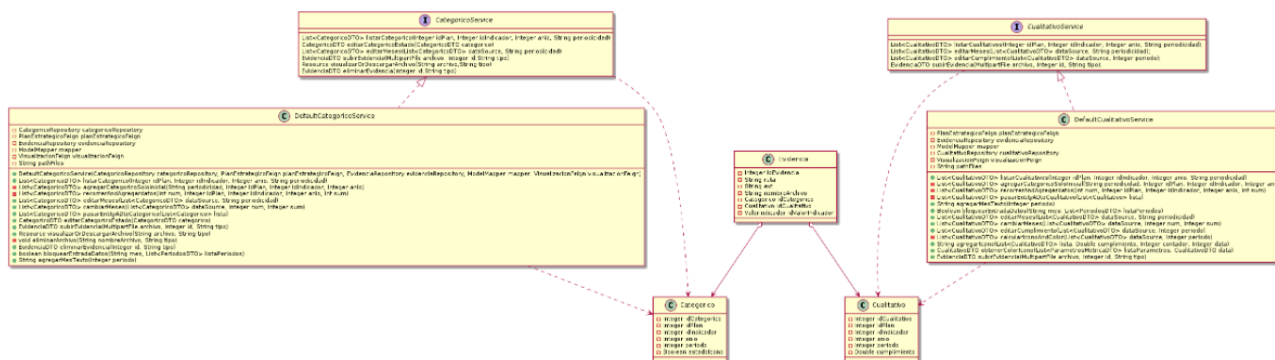


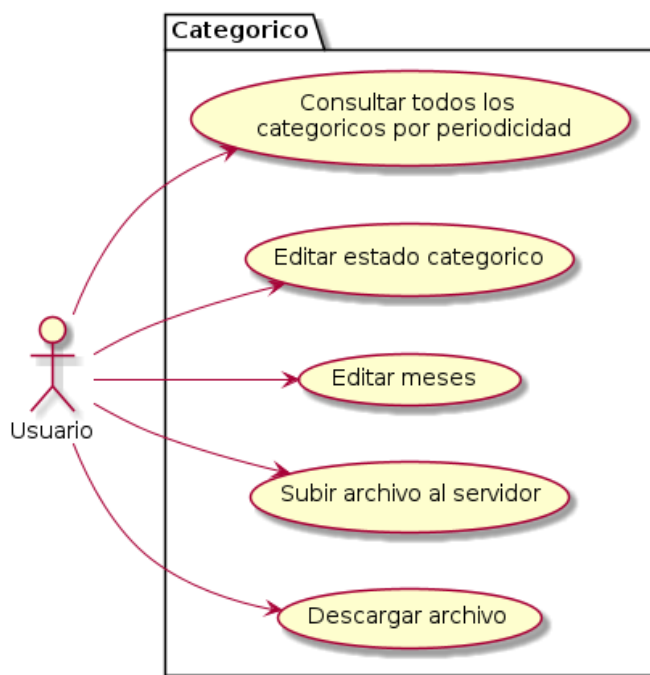
Ilustración 25, Diagrama de clases de Compromisos



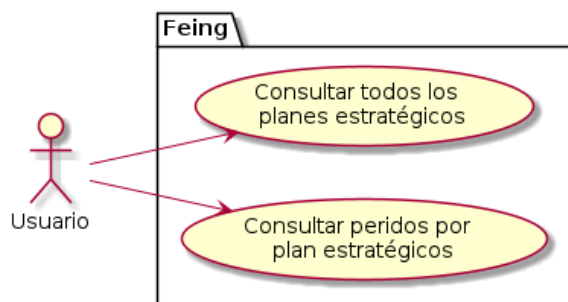
**Registrar-valores-back**, módulo encargado de la gestión de los valores categóricos y la generación de evidencias, visualizar o descargar archivos del servidor, gestión de evidencias en la base de datos, gestión de cumplimientos de valores cualitativos, generar evidencias cualitativas, gestión de las metas y de los indicadores.



*Ilustración 28, Diagrama de clases del módulo Registrar-valores-back*

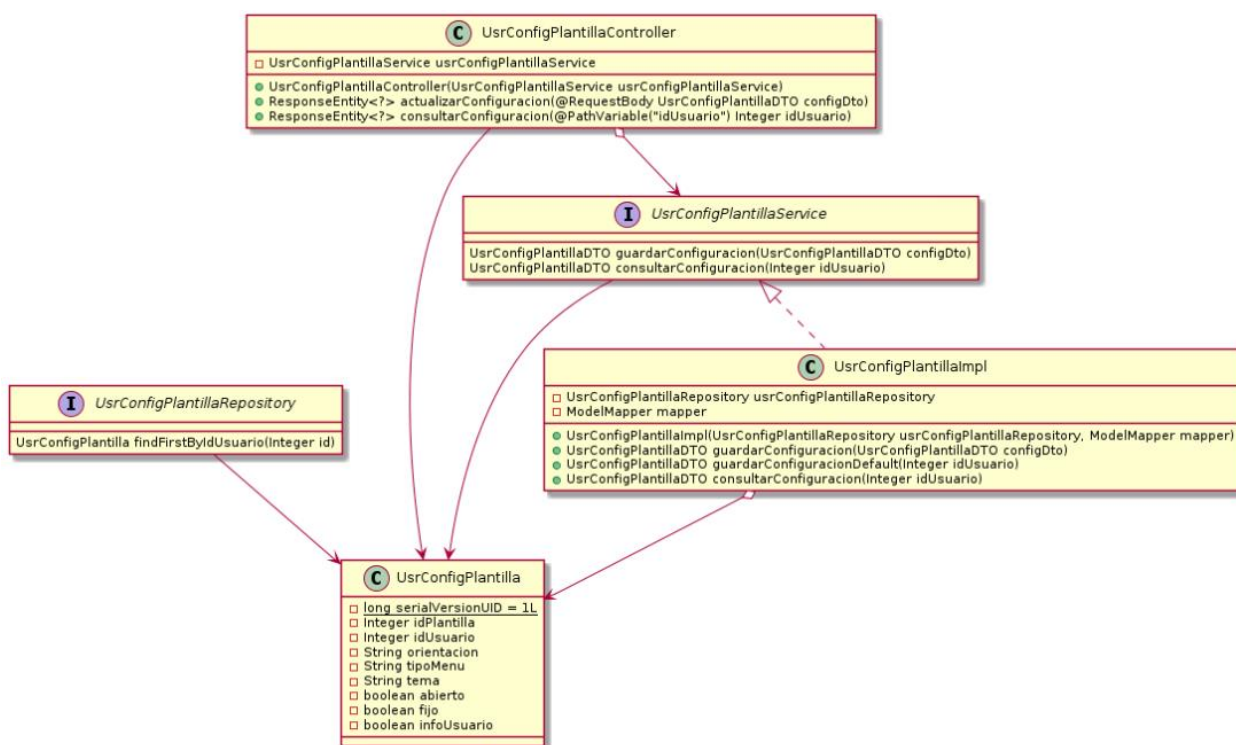


*Ilustración 29, Diagrama de casos de uso de Valores Categóricos*



*Ilustración 30, Diagrama de casos de uso de Feing*

**Usuarios**, Módulo encargado de consultar y actualizar la configuración de las plantillas de un usuario, la gestión de los roles y la gestión de los usuarios en la base de datos.



*Ilustración 31, Diagrama de clases 1 de módulo Usuarios*

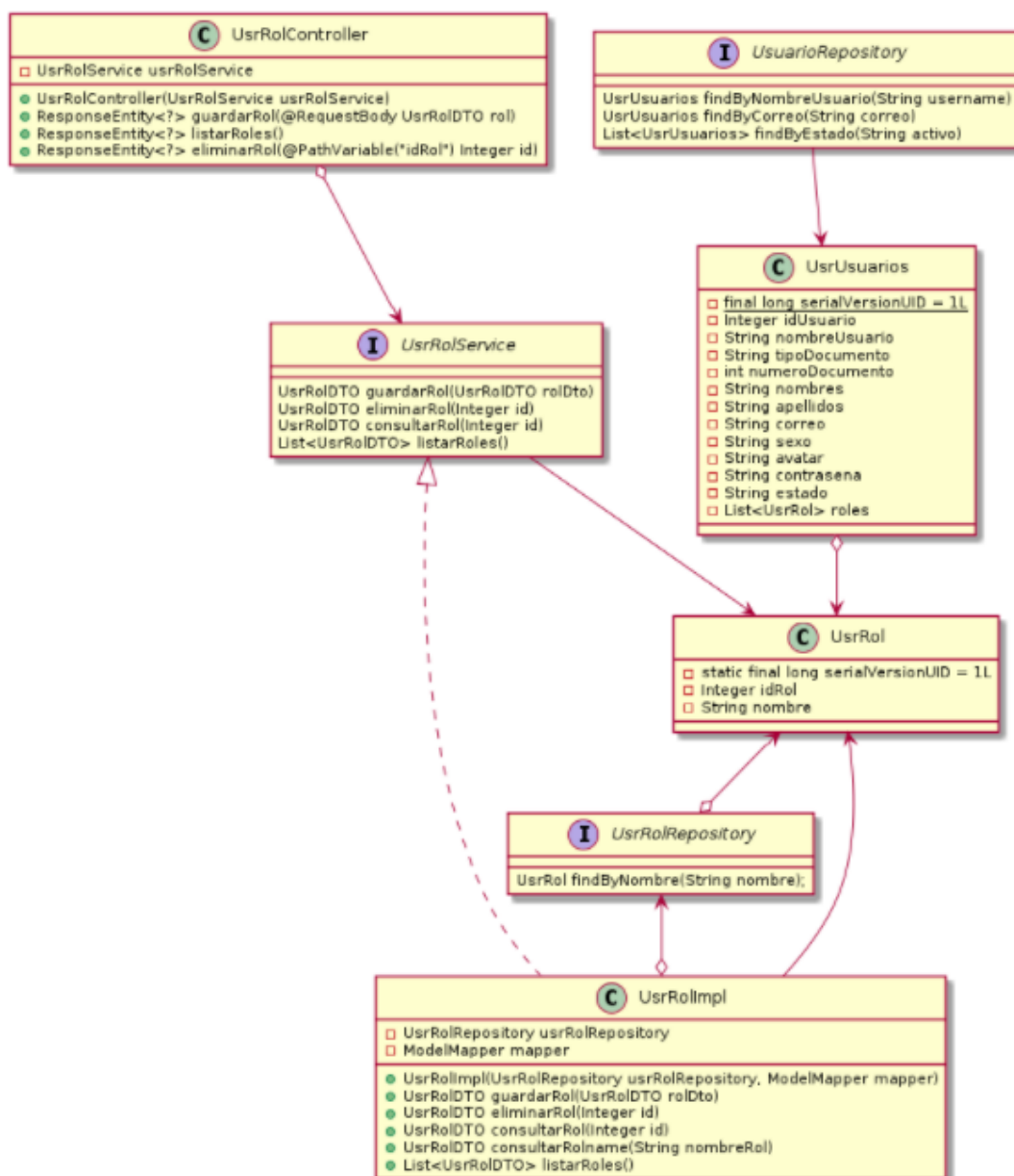


Ilustración 32, Diagrama de clases 2 de módulo Usuarios



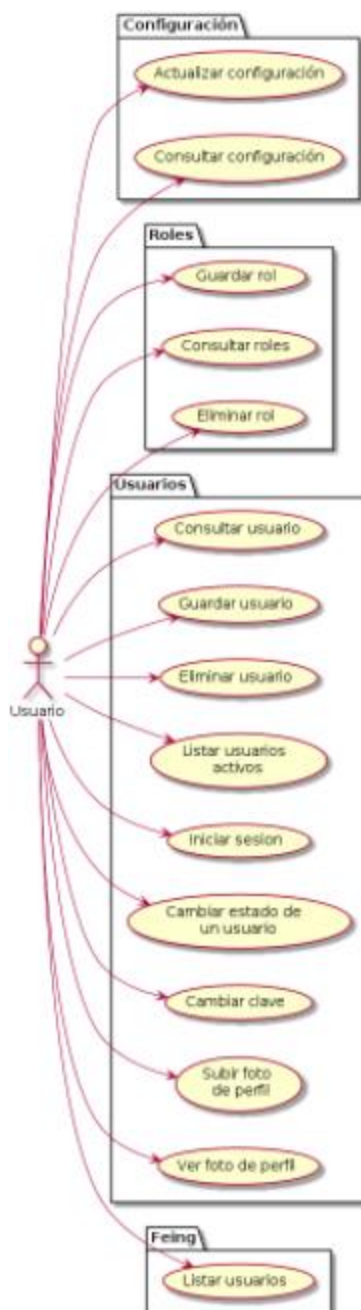
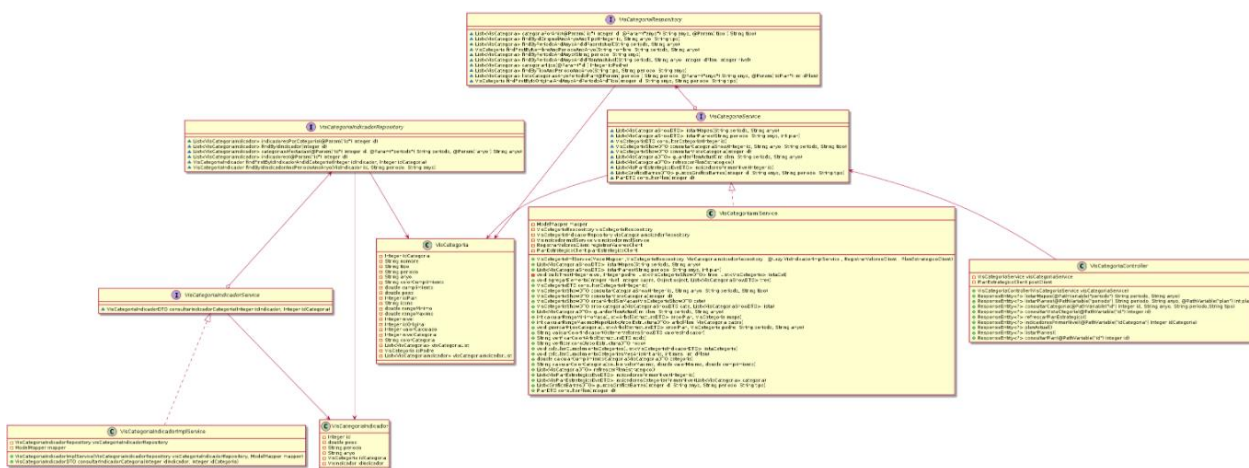


Ilustración 33, Diagrama de casos de uso de módulo Usuarios

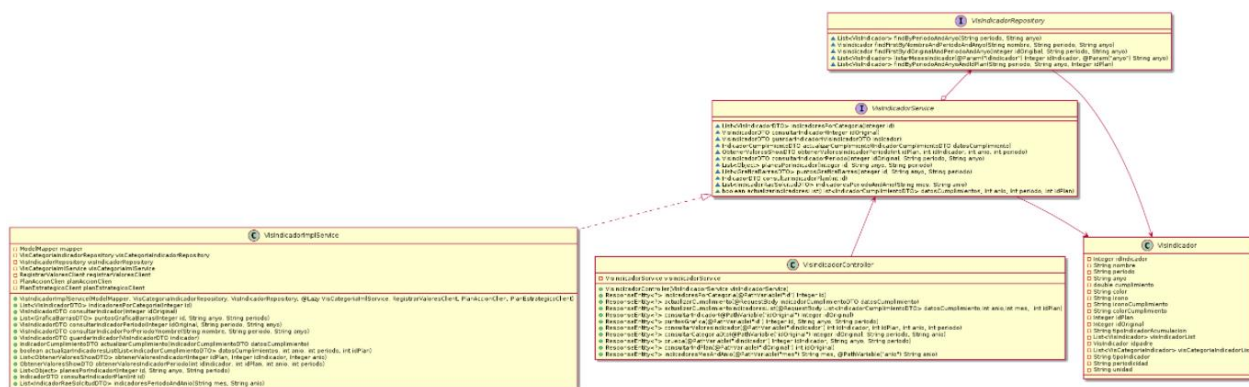


**Visualización**, módulo encargado de la gestión estratégica utilizada para definir y hacer seguimiento a la estrategia de una organización.

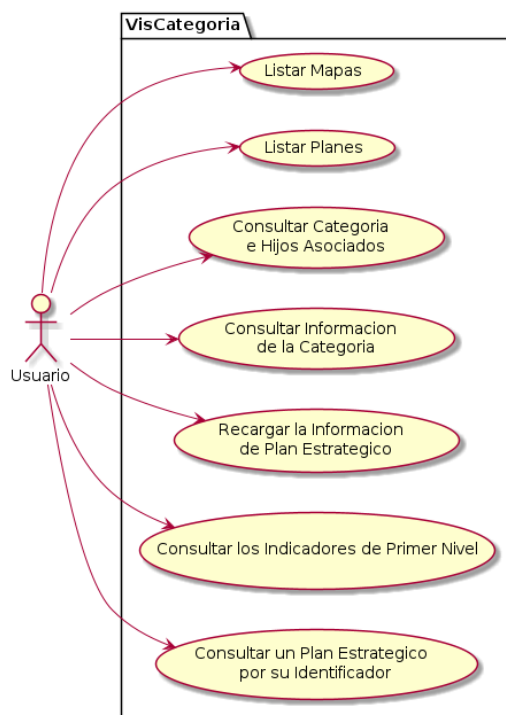
A su vez, BSC cuenta con unos propósitos tales cómo, describir y comunicar su estrategia, medir una estrategia en específico y hacer un seguimiento de las acciones que se están tomando para mejorar los resultados.



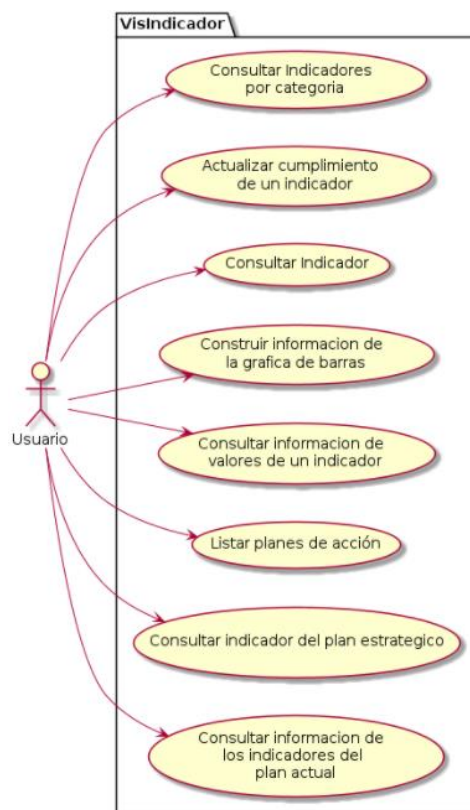
*Ilustración 34, Diagrama de clases Vis categoría del módulo de Visualización*



*Ilustración 35, Diagrama de clases Vis indicador del módulo de Visualización*



*Ilustración 36, Diagrama de casos de uso de Vis categoría del módulo de Visualización*



*Ilustración 37, Diagrama de casos de uso de Vis indicador del módulo de Visualización*

## 8. Bibliografía

Encarna Abellán. (2020). *Scrum: qué es y cómo funciona esta metodología*.

<https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html>

Píxel web pr. (2018), *Fundamentos de Scrum, metodología ágil gestión de proyectos*.

<https://www.udemy.com/fundamentos-de-scrum-metodologia-agil-gestion-de-proyectos>

Daniel Jarque. (2019), *Qué es un framework*. [https://www.makingscience.com/blog/que-es-](https://www.makingscience.com/blog/que-es-un-framework/)

[un-framework/](https://www.makingscience.com/blog/que-es-un-framework/)

Plantuml. (2021). *Guía De Referencia Del Lenguaje Plantuml*.

[http://pdf.plantuml.net/PlantUML\\_Language\\_Reference\\_Guide\\_es.pdf](http://pdf.plantuml.net/PlantUML_Language_Reference_Guide_es.pdf)

Keir Whitaker (2016). *What is Markdown Syntax?*.

<https://www.shopify.com/partners/blog/86763270-what-is-markdown-syntax>

Apache NetBeans. (2021). *Apache NetBeans History*.

<http://netbeans.apache.org/about/history.html>

Javier Cristobal (2016), *¿Qué es Markdown?*. <https://markdown.es/>

Spring. (2021). *Why Spring?*. <https://Spring.Io/Why-Spring>